

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

**IEEE Xplore®**
RELEASE 1.8Welcome
United States Patent and Trademark Office

» See

[Help](#) [FAQ](#) [Terms](#) [IEEE Peer Review](#)[Quick Links](#)

Welcome to IEEE Xplore®

- ☐ Home
- ☐ What Can I Access?
- ☐ Log-out

Tables of Contents

- ☐ Journals & Magazines
- ☐ Conference Proceedings
- ☐ Standards

Search

- ☐ By Author
- ☐ Basic
- ☐ Advanced

Member Services

- ☐ Join IEEE
- ☐ Establish IEEE Web Account
- ☐ Access the IEEE Member Digital Library

IEEE Enterprise

- ☐ Access the IEEE Enterprise File Cabinet

Print Format

[Home](#) | [Log-out](#) | [Journals](#) | [Conference Proceedings](#) | [Standards](#) | [Search by Author](#) | [Basic Search](#) | [Advanced Search](#) | [Join IEEE](#) | [Web Account](#) | [New this week](#) | [OPAC Linking Information](#) | [Your Feedback](#) | [Technical Support](#) | [Email Alerting](#) | [No Robots Please](#) | [Release Notes](#) | [IEEE Online Publications](#) | [Help](#) | [FAQ](#) | [Terms](#) | [Back to Top](#)

Your search matched **0** of **1047691** documents.A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance Descending** order.**Refine This Search:**

You may refine your search by editing the current search expression or entering a new one in the text box.

☐ Check to search within this result set**Results Key:****JNL** = Journal or Magazine **CNF** = Conference **STD** = Standard**Results:****No documents matched your query.**

**IEEE Xplore®**
RELEASE 1.8Welcome
United States Patent and Trademark Office

» Adva

[Help](#) [FAQ](#) [Terms](#) [IEEE Peer Review](#)[Quick Links](#)

Welcome to IEEE Xplore®

- ☐ Home
- ☐ What Can I Access?
- ☐ Log-out

Tables of Contents

- ☐ Journals & Magazines
- ☐ Conference Proceedings
- ☐ Standards

Search

- ☐ By Author
- ☐ Basic
- ☐ Advanced

Member Services

- ☐ Join IEEE
- ☐ Establish IEEE Web Account
- ☐ Access the IEEE Member Digital Library

IEEE Enterprise

- ☐ Access the IEEE Enterprise File Cabinet

Try our New Full-text Search Prototype **GO**[Help](#)

- 1) Enter a single keyword, phrase, or Boolean expression.
Example: acoustic imaging (means the phrase acoustic imaging plus any stem variations)
- 2) Limit your search by using search operators and field codes, if desired.
Example: optical <and> (fiber <or> fibre) <in> ti
- 3) Limit the results by selecting Search Options.
- 4) Click Search. See [Search Examples](#)

```
(write <near/2> (delay$ or  
stall$)) and ((write and  
read) <sentence> ((same or  
identical) <near/2> (address
```

Start Search

Clear

Note: This function returns plural and suffixed forms of the keyword(s).

Search operators: <and> <or> <not> <in> [More](#)

Field codes: au (author), ti (title), ab (abstract), jn (publication name), de (index term) [More](#)

Search Options:

Select publication types:

- ☒ IEEE Journals
- ☒ IEE Journals
- ☒ IEEE Conference proceedings
- ☒ IEE Conference proceedings
- ☒ IEEE Standards

Select years to search:

From year: to

Organize search results by:

Sort by: In: orderList Results per page



[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

Search: ☒ The ACM Digital Library ☐ The Guide

(write <near/2> (delay* or stall*)) and ((write and read) <sen



THE ACM DIGITAL LIBRARY

[Feedback](#) [Report a problem](#) [Satisfaction sur](#)

Terms used

write near/2 delay or stall and **write** and **read sentence same** or **identical near/2 address** or **line** or **location**

1

Sort results by

[Save results to a Binder](#)

[Try an Advanced Search](#)

Display results

[Search Tips](#)

[Try this search in The ACM Guide](#)

☐ [Open results in a new window](#)

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale ☐ ☐

1 [Fast detection of communication patterns in distributed executions](#)

Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**

Full text available: pdf(4.21 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

2 [Translator writing systems](#)

Jerome Feldman, David Gries

February 1968 **Communications of the ACM**, Volume 11 Issue 2

Full text available: pdf(4.47 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

A critical review of recent efforts to automate the writing of translators of programming languages is presented. The formal study of syntax and its application to translator writing are discussed in Section II. Various approaches to automating the postsyntactic (semantic) aspects of translator writing are discussed in Section III, and several related topics in Section IV.

Keywords: compiler compiler-compiler, generator, macroprocessor, meta-assembler, metacompiler, parser, semantics, syntactic analysis, syntax, syntax-directed, translator, translator writing system

3 [Pen computing: a technology overview and a vision](#)

André Meyer

July 1995 **ACM SIGCHI Bulletin**, Volume 27 Issue 3

Full text available: pdf(5.14 MB)


Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

This work gives an overview of a new technology that is attracting growing interest in public as we as in the computer industry itself. The visible difference from other technologies is in the use of a pen or pencil as the primary means of interaction between a user and a machine, picking up the familiar pen and paper interface metaphor. From this follows a set of consequences that will be analyzed and put into context with other emerging technologies and visions. Starting with a short historic ...

4 [Reading and writing with computers: a framework for explaining differences in performance](#)

Wilfred . nsen, Christin s

August 1988 **Communications of the ACM**, Volume 31 Issue 9

Full text available:  pdf(1.40 MB)


Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Several factors can influence the behavior of users as they read and write with computers. Recent research indicates that both quality and quantity depend upon page size, legibility, responsiveness and tangibility.

5 Technique for automatically correcting words in text

Karen Kukich

December 1992 **ACM Computing Surveys (CSUR)**, Volume 24 Issue 4

Full text available:  pdf(6.23 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)


Research aimed at correcting words in text has focused on three progressively more difficult problems: (1) nonword error detection; (2) isolated-word error correction; and (3) context-dependent word correction. In response to the first problem, efficient pattern-matching and n-gram analysis techniques have been developed for detecting strings that do not appear in a given word list. In response to the second problem, a variety of general and application-specific spelling cor ..

Keywords: n-gram analysis, Optical Character Recognition (OCR), context-dependent spelling correction, grammar checking, natural-language-processing models, neural net classifiers, spell checking, spelling error detection, spelling error patterns, statistical-language models, word recognition and correction

6 Interactive Editing Systems: Part II

Norman Meyrowitz, Andries van Dam

September 1982 **ACM Computing Surveys (CSUR)**, Volume 14 Issue 3

Full text available:  pdf(9.17 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

7 A Mechanical Proof Procedure and its Realization in an Electronic Computer

Dag Prawitz, Haåkan Prawitz, Neri Voghera


April 1960 **Journal of the ACM (JACM)**, Volume 7 Issue 2

Full text available:  pdf(1.57 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

8 FORTRAN vs. Basic FORTRAN: a programming language for informational processing on automatic data processing systems

October 1964 **Communications of the ACM**, Volume 7 Issue 10

Full text available:  pdf(3.90 MB)

Additional Information: [full citation](#)

9 Abstract state machines capture parallel algorithms

Andreas Blass, Yuri Gurevich

October 2003 **ACM Transactions on Computational Logic (TOCL)**, Volume 4 Issue 4

Full text available:  pdf(610.28 KB)

Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)



We give an axiomatic description of parallel, synchronous algorithms. Our main result is that even such algorithm can be simulated, step for step, by an abstract state machine with a background that provides for multisets.

Keywords: ASM thesis, Parallel algorithm, abstract state machine, postulates for parallel computation

10 The FINITE STRING Newsletter: Abstracts of current literature

Computational Linguistics Staff

January 1987 **Computational Linguistics**, Volume 13 Issue 1-2


Full text available:  pdf(6.15 MB) 
[Publisher Site](#)

Additional Information: [full citation](#)

11 Document Formatting Systems: Survey, Concepts, and Issues

Richard Furuta, Jeffrey Scofield, Alan Shaw

September 1982 **ACM Computing Surveys (CSUR)**, Volume 14 Issue 3

Full text available:  pdf(5.36 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

12 Sheaved memory: architectural support for state saving and restoration in pages systems

M. E. Staknis

April 1989 **ACM SIGARCH Computer Architecture News , Proceedings of the third international conference on Architectural support for programming languages and operating systems**, Volume 17 Issue 2

Full text available:  pdf(973.26 KB)


Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The concept of read-one/write-many paged memory is introduced and given the name sheaved memory. It is shown that sheaved memory is useful for efficiently maintaining checkpoints in main memory and for providing state saving and state restoration for software that includes recovery blocks or similar control structures. The organization of sheaved memory is described in detail, and a design is presented for a prototype sheaved-memory module that can be built easily from inext.

13 Clarification of Fortran standards—second report

C. Kerpelman

October 1971 **Communications of the ACM**, Volume 14 Issue 10

Full text available:  pdf(1.84 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)


In 1966, after four years of effort, Fortran became the first programming language standardized in the United States. Since that initial achievement, study and application of the standard specifications have revealed the need for maintenance of the standards. As the result of work initiated in 1967, an initial set of clarifying interpretations was prepared and this clarification was published in Communications of the ACM in May 1969. That work has continued and has resulted in the preparati ...

Keywords: American National Standard, Basic Fortran, Fortran, language standard clarification, language standard interpretation, language standard maintenance, language standard specification, programming language, standardization, standardization committee

14 An Elementary Discussion of Compiler/Interpreter Writing

R. L. Glass

January 1969 **ACM Computing Surveys (CSUR)**, Volume 1 Issue 1

Full text available:  pdf(1.85 MB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

15 Status report of the graphic standards planning committee of ACM/SIGGRAPH: State-of-the-art of graphic software packages

Computer Graphics staff

September 1977 **ACM SIGGRAPH Computer Graphics**, Volume 11 Issue 3

Full text available:  pdf(9.03 MB)

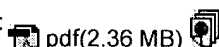
Additional Information: [full citation](#), [references](#)

16 Special issue on natural language generation: Generating natural language summaries from multiple on-line sources

Dragomir R. Radev, Kathleen R. McKeown

September 1998 **Computational Linguistics**, Volume 24 Issue 3

Full text available:



[Publisher Site](#)


Additional Information: [full citation](#), [abstract](#), [references](#)

We present a methodology for summarization of news about current events in the form of briefing that include appropriate background (historical) information. The system that we developed, SUMMONS, uses the output of systems developed for the DARPA Message Understanding Conferences to generate summaries of multiple documents on the same or related events, presenting similarities and differences, contradictions, and generalizations among sources of information. We describe the various components ...

17 Flexible Diff-ing in a collaborative writing system

Christine M. Neuwirth, Ravinder Chandhok, David S. Kaufer, Paul Erion, James Morris, Dale Miller

December 1992 **Proceedings of the 1992 ACM conference on Computer-supported cooperative work**

Full text available:  pdf(946.01 KB)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

Keywords: collaborative writing, flexible differencing, text comparison

18 Talking to UNIX in English: an overview of UC

Robert Wilensky, Yigal Arens, David Chin

June 1984 **Communications of the ACM**, Volume 27 Issue 6

Full text available:  pdf(2.03 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)


UC is a natural language help facility which advises users in using the UNIX operating system. Users can query UC about how to do things, command names and formats, online definitions of UNIX or general operating systems terminology, and debugging problems in using commands. UC is comprised of the following components: a language analyzer and generator, a context and memory model, an experimental common-sense planner, highly extensible knowledge bases on both the UNIX domain and the ...

Keywords: ellipsis, goal analysis, memory models, natural dialogue, reference disambiguation

19 The nested rectangular array as a model of data

Trenchard More

May 1979 **ACM SIGAPL APL Quote Quad , Proceedings of the international conference on APL: part 1**, Volume 9 Issue 4

Full text available:  pdf(2.11 MB)


Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Data, like electricity and gravity, are part of the world in which we live. Some occur naturally, as in the genetic code, while most occur as a consequence of language and social organization. The search for a theory of data, which begins with the choice of a model, is as important and interesting as the development of theories in physics, economics, and psychology. Most models of data are collections, such as the unnested array of APL, the one-axis nested list of LISP, and the s ...

20 Parallel execution of prolog programs: a survey

Gopal Gupta, Enrico Pontelli, Khayri A.M. Ali, Mats Carlsson, Manuel V. Hermenegildo

July 2001 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 23 Issue 4

Full text available:  pdf(1.95 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Since the early days of logic programming, researchers in the field realized the potential for exploitation of parallelism present in the execution of logic programs. Their high-level nature, the presence of nondeterminism, and their referential transparency, among other characteristics, make logic programs interesting candidates for obtaining speedups through parallel execution. At the

Results (page 1): (write <near/2> (delay* or stall*)) and ((write and read) <sentence> ((sa... Page 5 of 5

same time, the fact that the typical applications of logic programming frequently involve irregular computatio ...

Keywords: Automatic parallelization, constraint programming, logic programming, parallelism, prolog

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2004 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)